Atmel - ATUC128D4-AUT Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	AVR
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	35
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	1.65V ~ 3.6V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/atuc128d4-aut

Table 3-1. Multiplexed Signals on I/O Pins

48-pin Package	64-pin Package	PIN	GPIO	Supply	Pad Type	GPIO Function				Other Functions
						A	B	C	D	
9	11	PA05	5	VDDANA	Analog I/O	EIC - EXTINT[8]	PKGANA - ADCIN2	GLOC - OUT[1]	CAT - CSB[3]	
10	12	PA06	6	VDDANA	Analog I/O	EIC - EXTINT[1]	PKGANA - ADCIN3	GLOC - IN[7]	CAT - CSA[4]	
11	13	PA07	7	VDDANA	Analog I/O	PWMA - PWMA[0]	PKGANA - ADCIN4	GLOC - IN[8]	CAT - CSB[4]	
12	14	PA08	8	VDDANA	Analog I/O	PWMA - PWMA[1]	PKGANA - ADCIN5	GLOC - IN[9]	CAT - CSA[5]	
20	28	PA09	9	VDDIO	Normal I/O, 5V tolerant	TWIMS - TWCK	SPI - NPCS[2]	USART1 - CTS	CAT - CSB[5]	
21	29	PA10	10	VDDIO	Normal I/O, 5V tolerant	TWIMS - TWD	SPI - NPCS[3]	USART1 - RTS	CAT - CSA[6]	
22	30	PA11	11	VDDIO	Normal I/O	USART0 - RTS	TC - A2	PWMA - PWMA[0]	CAT - CSB[6]	OSC32 - XIN
23	31	PA12	12	VDDIO	Normal I/O	USART0 - CTS	TC - B2	PWMA - PWMA[1]	CAT - CSA[7]	OSC32 - XOUT
25	33	PA13	13	VDDIO	Normal I/O	EIC - EXTINT[0]	PWMA - PWMA[2]	USART0 - CLK	CAT - CSB[7]	
26	34	PA14	14	VDDIO	Normal I/O	SPI - MOSI	PWMA - PWMA[3]	EIC - EXTINT[2]	CAT - CSA[8]	
27	35	PA15	15	VDDIO	Normal I/O	SPI - SCK	PWMA - PWMA[4]	USART2 - CLK	CAT - CSB[8]	
28	36	PA16	16	VDDIO	Normal I/O	SPI - NPCS[0]	TC - CLK1	PWMA - PWMA[4]	CAT - CSA[9]	
29	37	PA17	17	VDDIO	Normal I/O	SPI - NPCS[1]	TC - CLK2	SPI - SCK	CAT - CSB[9]	
30	39	PA18	18	VDDIO	Normal I/O	USART0 - RXD	PWMA - PWMA[5]	SPI - MISO	CAT - CSA[10]	OSC0 - XIN
31	40	PA19	19	VDDIO	Normal I/O	USART0 - TXD	PWMA - PWMA[6]	SPI - MOSI	CAT - CSB[10]	OSC0 - XOUT
32	44	PA20	20	VDDIO	Normal I/O	USART1 - CLK	TC - CLK0	USART2 - RXD	CAT - CSA[11]	
33	45	PA21	21	VDDIO	Normal I/O	PWMA - PWMA[2]	TC - A1	USART2 - TXD	CAT - CSB[11]	
34	46	PA22	22	VDDIO	Normal I/O	PWMA - PWMA[6]	TC - B1	ADCIFD - EXTTRIG	CAT - CSA[12]	
35	47	PA23	23	VDDIO	Normal I/O	USART1 - TXD	SPI - NPCS[1]	EIC - EXTINT[3]	CAT - CSB[12]	
43	59	PA24	24	VDDIO	Normal I/O	USART1 - RXD	SPI - NPCS[0]	EIC - EXTINT[4]	CAT - CSB[15]	
44	60	PA25	25	VDDIO	Normal I/O	SPI - MISO	PWMA - PWMA[3]	EIC - EXTINT[5]	CAT - CSA[16]	
45	61	PA26	26	VDDIO	Normal I/O	IISC - IWS	USART2 - TXD	TC - A0	CAT - CSB[16]	
46	62	PA27	27	VDDIO	Normal I/O	IISC - ISCK	USART2 - RXD	TC - B0	CAT - CSA[0]	
	41	PA28	28	VDDIO	Normal I/O	USART0 - CLK	PWMA - PWMA[4]	SPI - MISO	CAT - CSB[21]	
	42	PA29	29	VDDIO	Normal I/O	TC - CLK0	TC - CLK1	SPI - MOSI	CAT - CSA[22]	
	15	PA30	30	VDDANA	Analog I/O	PKGANA - ADCIN6	EIC - EXTINT[6]	SCIF - GCLK[2]	CAT - CSA[18]	
	16	PA31	31	VDDANA	Analog I/O	PKGANA - ADCIN7	EIC - EXTINT[7]	PWMA - PWMA[6]	CAT - CSB[18]	
	6	PB00	32	VDDIO	Normal I/O	TC - A0	EIC - EXTINT[4]	USART2 - CTS	CAT - CSA[17]	
	7	PB01	33	VDDIO	Normal I/O	TC - B0	EIC - EXTINT[5]	USART2 - RTS	CAT - CSB[17]	
	24	PB02	34	VDDIO	Normal I/O	EIC - EXTINT[6]	TC - A1	USART1 - TXD	CAT - CSA[19]	
	25	PB03	35	VDDIO	Normal I/O	EIC - EXTINT[7]	TC - B1	USART1 - RXD	CAT - CSB[19]	
	26	PB04	36	VDDIO	Normal I/O	USART1 - CTS	SPI - NPCS[3]	TC - CLK2	CAT - CSA[20]	
	27	PB05	37	VDDIO	Normal I/O	USART1 - RTS	SPI - NPCS[2]	PWMA - PWMA[5]	CAT - CSB[20]	
	38	PB06	38	VDDIO	Normal I/O	IISC - ISCK	PWMA - PWMA[5]	GLOC - IN[15]	CAT - CSA[21]	
	43	PB07	39	VDDIO	Normal I/O	IISC - ISDI	EIC - EXTINT[2]	GLOC - IN[11]	CAT - CSB[22]	
	54	PB08	40	VDDIO	Normal I/O	IISC - IWS	EIC - EXTINT[0]	GLOC - IN[14]	CAT - CSA[23]	
	55	PB09	41	VDDIO	Normal I/O	IISC - ISCK	IISC - IMCK	GLOC - IN[3]	CAT - CSB[23]	
	57	PB10	42	VDDIO	Normal I/O	IISC - ISDO	TC - A2	USART0 - RXD	CAT - CSA[24]	
	58	PB11	43	VDDIO	Normal I/O	IISC - IWS	TC - B2	USART0 - TXD	CAT - CSB[24]	

Table 3-1. Multiplexed Signals on I/O Pins

48-pin Package	64-pin Package	PIN	GPIO	Supply	Pad Type	GPIO Function				Other Functions
						A	B	C	D	
2	2	PB12	44	VDDIO	Normal I/O	SPI - NPCS[0]	IISC - IMCK	GLOC - OUT[0]		JTAG-TCK
6	8	PB13	45	VDDIO	Normal I/O	CAT - SYNC	SCIF - GCLK[2]	GLOC - IN[4]	CAT - CSA[2]	
38	50	PB14	46	VDDIO	Normal I/O	USBC - DP	USART0 - RXD	GLOC - OUT[2]	CAT - CSA[13]	
39	51	PB15	47	VDDIO	Normal I/O	USBC - DM	USART0 - TXD	GLOC - IN[12]	CAT - CSB[13]	
40	52	PB16	48	VDDIO	Input only, 5V tolerant	USBC - VBUS		GLOC - IN[10]		USB-VBUS
41	53	PB17	49	VDDIO	Normal I/O	IISC - ISDO	USART0 - RTS	GLOC - IN[13]		
42	56	PB18	50	VDDIO	Normal I/O	IISC - ISDI	CAT - SYNC	GLOC - OUT[3]	CAT - CSA[15]	

See Section 4. for a description of the various peripheral signals.

Refer to ”Electrical Characteristics” on page 37 for a description of the electrical properties of the pad types used.

3.2.2 Peripheral Functions

Each GPIO line can be assigned to one of several peripheral functions. The following table describes how the various peripheral functions are selected. The last listed function has priority in case multiple functions are enabled.

Table 3-2. Peripheral Functions

Function	Description
A	GPIO peripheral selection A
B	GPIO peripheral selection B
C	GPIO peripheral selection C
D	GPIO peripheral selection D

3.2.3 JTAG Port Connections

If the JTAG is enabled, the JTAG will take control over a number of pins, irrespective of the I/O Controller configuration.

Table 3-3. JTAG Pinout

48-pin or 64-pin Package	Pin Name	JTAG Pin
2	PB12	TCK
5	PA02	TMS
4	PA01	TDO
3	PA00	TDI

4. Signal Descriptions

The following table gives details on signal name classified by peripheral.

Table 4-1. Signal Descriptions List

Signal Name	Function	Type	Active Level	Comments
aWire - AW				
DATA	aWire data	I/O		
DATAOUT	aWire data output for 2-pin mode	I/O		
External Interrupt Controller - EIC				
NMI	Non-Maskable Interrupt	Input		
EXTINT8 - EXTINT1	External interrupt	Input		
JTAG module - JTAG				
TCK	Test Clock	Input		
TDI	Test Data In	Input		
TDO	Test Data Out	Output		
TMS	Test Mode Select	Input		
Power Manager - PM				
RESET_N	Reset	Input	Low	
Basic Pulse Width Modulation Controller - PWMA				
PWMA6 - PWMA0	PWMA channel waveforms	Output		
System Control Interface - SCIF				
GCLK2 - GCLK0	Generic clock	Output		
XIN0	Oscillator 0 XIN Pin	Analog		
XOUT0	Oscillator 0 XOUT Pin	Analog		
XIN32	32K Oscillator XIN Pin	Analog		
XOUT32	32K Oscillator XOUT Pin	Analog		
Serial Peripheral Interface - SPI				
MISO	Master In Slave Out	I/O		
MOSI	Master Out Slave In	I/O		
NPCS3 - NPCS0	SPI Peripheral Chip Select	I/O	Low	
SCK	Clock	I/O		
Timer/Counter - TC				
A0	Channel 0 Line A	I/O		
A1	Channel 1 Line A	I/O		

Table 4-1. Signal Descriptions List

EXTTRIG	ADCIFD EXTTRIG	Input		
AD7 - AD0	ADC Inputs	Analog		
Power				
VDDIO	Digital I/O Power Supply	Power Input		3.0 V to 3.6V.
VDDANA	Analog Power Supply	Power Input		3.0 V to 3.6V
ADVREF	Analog Reference Voltage	Power Input		2.6 V to 3.6 V
VDDCORE	Core Power Supply	Power Input		1.65 V to 1.95 V
VDDIN	Voltage Regulator Input	Power Input		3.0 V to 3.6V
VDDOUT	Voltage Regulator Output	Power Output		1.65 V to 1.95V
GNDANA	Analog Ground	Ground		
GND	Ground	Ground		
General Purpose I/O pin - GPIOA, GPIOB				
PA31 - PA00	General Purpose I/O Controller GPIO A	I/O		
PB18 - PB00	General Purpose I/O Controller GPIO B	I/O		

4.1 I/O Line Considerations

4.1.1 JTAG Pins

The JTAG is enabled if TCK is low while the RESET_N pin is released. The TCK, TMS, and TDI pins have pull-up resistors when JTAG is enabled. TDO pin is an output, driven at VDDIO, and has no pull-up resistor. These JTAG pins can be used as GPIO pins and muxed with peripherals when the JTAG is disabled.

4.1.2 RESET_N Pin

The RESET_N pin is a schmitt input and integrates a programmable pull-up resistor to VDDIO. As the product integrates a power-on reset detector, the RESET_N pin can be left unconnected in case no reset from the system needs to be applied to the product.

The RESET_N pin is also used for the aWire debug protocol. When the pin is used for debugging, it must not be driven by the application.

4.1.3 TWI Pins

When these pins are used for TWI, the pins are open-drain outputs with slew-rate limitation and inputs with inputs with spike-filtering. When used as GPIO pins or used for other peripherals, the pins have the same characteristics as GPIO pins.

4.1.4 GPIO Pins

All the I/O lines integrate a pull-up resistor. Programming of this pull-up resistor is performed

5. Processor and Architecture

Rev: 2.1.2.0

This chapter gives an overview of the AVR32UC CPU. AVR32UC is an implementation of the AVR32 architecture. A summary of the programming model, and instruction set is presented. For further details, see the *AVR32 Architecture Manual* and the *AVR32UC Technical Reference Manual*.

5.1 Features

- **32-bit load/store AVR32A RISC architecture**
  - **15 general-purpose 32-bit registers**
  - **32-bit Stack Pointer, Program Counter and Link Register reside in register file**
  - **Fully orthogonal instruction set**
  - **Privileged and unprivileged modes enabling efficient and secure operating systems**
  - **Innovative instruction set together with variable instruction length ensuring industry leading code density**
  - **DSP extension with saturating arithmetic, and a wide variety of multiply instructions**
- **3-stage pipeline allowing one instruction per clock cycle for most instructions**
  - **Byte, halfword, word, and double word memory access**
  - **Multiple interrupt priority levels**

5.2 AVR32 Architecture

AVR32 is a new, high-performance 32-bit RISC microprocessor architecture, designed for cost-sensitive embedded applications, with particular emphasis on low power consumption and high code density. In addition, the instruction set architecture has been tuned to allow a variety of microarchitectures, enabling the AVR32 to be implemented as low-, mid-, or high-performance processors. AVR32 extends the AVR family into the world of 32- and 64-bit applications.

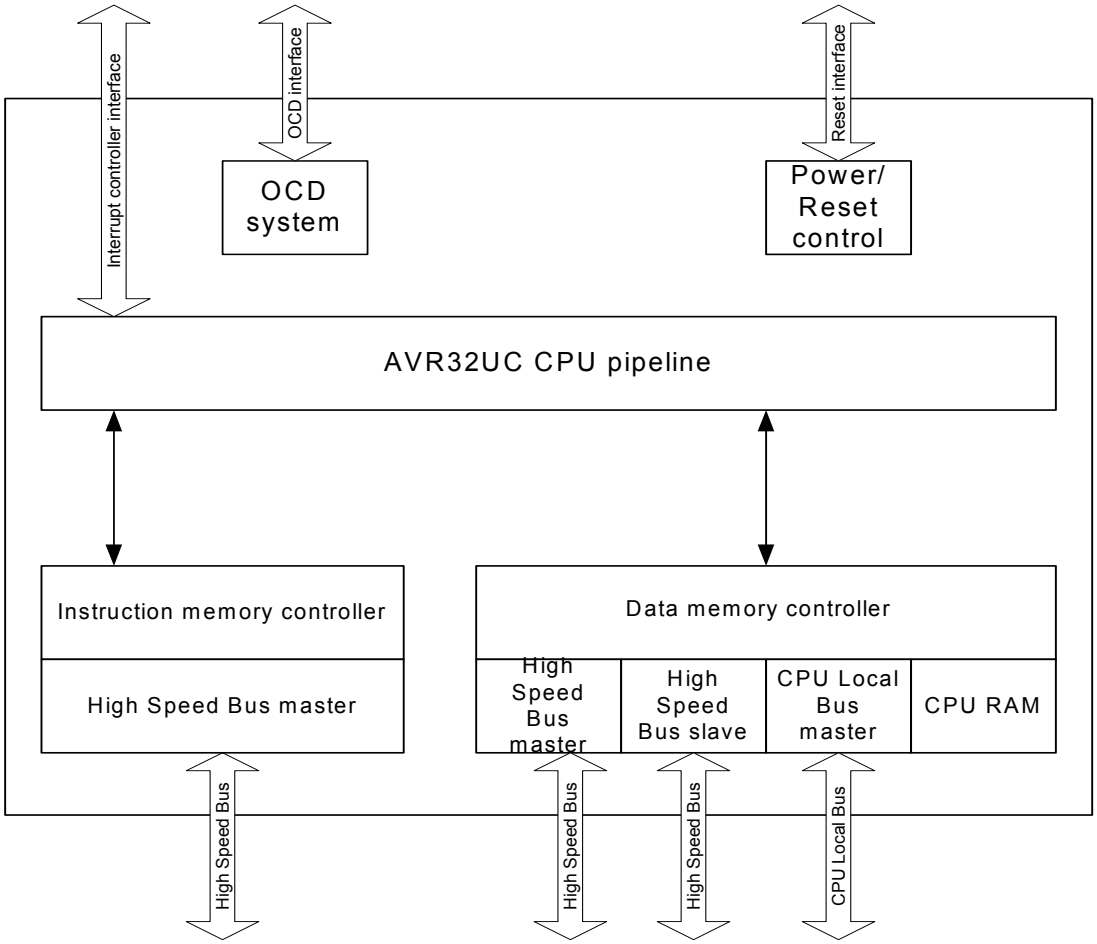
Through a quantitative approach, a large set of industry recognized benchmarks has been compiled and analyzed to achieve the best code density in its class. In addition to lowering the memory requirements, a compact code size also contributes to the core's low power characteristics. The processor supports byte and halfword data types without penalty in code size and performance.

Memory load and store operations are provided for byte, halfword, word, and double word data with automatic sign- or zero extension of halfword and byte data. The C-compiler is closely linked to the architecture and is able to exploit code optimization features, both for size and speed.

In order to reduce code size to a minimum, some instructions have multiple addressing modes. As an example, instructions with immediates often have a compact format with a smaller immediate, and an extended format with a larger immediate. In this way, the compiler is able to use the format giving the smallest code size.

Another feature of the instruction set is that frequently used instructions, like add, have a compact format with two operands as well as an extended format with three operands. The larger format increases performance, allowing an addition and a data move in the same instruction in a single cycle. Load and store instructions have several different formats in order to reduce code size and speed up execution.

Figure 5-1. Overview of the AVR32UC CPU



5.3.1 Pipeline Overview

AVR32UC has three pipeline stages, Instruction Fetch (IF), Instruction Decode (ID), and Instruction Execute (EX). The EX stage is split into three parallel subsections, one arithmetic/logic (ALU) section, one multiply (MUL) section, and one load/store (LS) section.

Instructions are issued and complete in order. Certain operations require several clock cycles to complete, and in this case, the instruction resides in the ID and EX stages for the required number of clock cycles. Since there is only three pipeline stages, no internal data forwarding is required, and no data dependencies can arise in the pipeline.

Figure 5-2 on page 23 shows an overview of the AVR32UC pipeline stages.

6. Memories

6.1 Embedded Memories

- **Internal High-Speed Flash**
  - **128Kbytes (ATUC128D)**
  - **64Kbytes (ATUC64D)**
    - **0 Wait State Access at up to 24 MHz in Worst Case Conditions**
    - **1 Wait State Access at up to 48 MHz in Worst Case Conditions**
    - **Pipelined Flash Architecture, allowing burst reads from sequential Flash locations, hiding penalty of 1 wait state access**
    - **100 000 Write Cycles, 15-year Data Retention Capability**
    - **4ms Page Programming Time, 8 ms Chip Erase Time**
    - **Sector Lock Capabilities, Bootloader Protection, Security Bit**
    - **32 Fuses, Erased During Chip Erase**
    - **User Page For Data To Be Preserved During Chip Erase**
- **Internal High-Speed SRAM, Single-cycle access at full speed**
  - **16Kbytes**

6.2 Physical Memory Map

The system bus is implemented as a bus matrix. All system bus addresses are fixed, and they are never remapped in any way, not even in boot. Note that AVR32UC CPU uses unsegmented translation, as described in the AVR32 Architecture Manual. The 32-bit physical address space is mapped as follows:

Table 6-1. UC3D Physical Memory Map

Device		Embedded SRAM	Embedded Flash	HSB-PB Bridge A	HSB-PB Bridge B
Start Address		0x0000_0000	0x8000_0000	0xFFFF_0000	0xFFFE_0000
Size	ATUC128D	16 Kbytes	128 Kbytes	64 Kbytes	64 Kbytes
	ATUC64D	16 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes

6.3 Peripheral Address Map

Table 6-2. Peripheral Address Mapping

Address		Peripheral Name
0xFFFE0000	USBC	USB 2.0 Interface - USBC
0xFFFE1000	HMATRIX	HSB Matrix - HMATRIX
0xFFFE1400	FLASHCDW	Flash Controller - FLASHCDW
0xFFFF0000	PDCA	Peripheral DMA Controller - PDCA
0xFFFF1000	INTC	Interrupt controller - INTC

Table 6-2. Peripheral Address Mapping

0xFFFF6400	GLOC	Glue Logic Controller - GLOC
0xFFFF6800	AW	aWire - AW

6.4 CPU Local Bus Mapping

Some of the registers in the GPIO module are mapped onto the CPU local bus, in addition to being mapped on the Peripheral Bus. These registers can therefore be reached both by accesses on the Peripheral Bus, and by accesses on the local bus.

Mapping these registers on the local bus allows cycle-deterministic toggling of GPIO pins since the CPU and GPIO are the only modules connected to this bus. Also, since the local bus runs at CPU speed, one write or read operation can be performed per clock cycle to the local bus-mapped GPIO registers.

The following GPIO registers are mapped on the local bus:

Table 6-3. Local Bus Mapped GPIO Registers

Port	Register	Mode	Local Bus Address	Access
A	Output Driver Enable Register (ODER)	WRITE	0x40000040	Write-only
		SET	0x40000044	Write-only
		CLEAR	0x40000048	Write-only
		TOGGLE	0x4000004C	Write-only
	Output Value Register (OVR)	WRITE	0x40000050	Write-only
		SET	0x40000054	Write-only
		CLEAR	0x40000058	Write-only
		TOGGLE	0x4000005C	Write-only
	Pin Value Register (PVR)	-	0x40000060	Read-only
B	Output Driver Enable Register (ODER)	WRITE	0x40000140	Write-only
		SET	0x40000144	Write-only
		CLEAR	0x40000148	Write-only
		TOGGLE	0x4000014C	Write-only
	Output Value Register (OVR)	WRITE	0x40000150	Write-only
		SET	0x40000154	Write-only
		CLEAR	0x40000158	Write-only
		TOGGLE	0x4000015C	Write-only
	Pin Value Register (PVR)	-	0x40000160	Read-only

Table 8-6. Normal I/O Pin Characteristics[1]

Symbol	Parameter	Condition		Min	Typ	Max	Units
I_{OL}	Output low-level current	V_{VDD} = 3.0V	[5]			4	mA
			[6]			8	mA
I_{OH}	Output high-level current	V_{VDD} = 3.0V	[5]			4	mA
			[6]			8	mA
F_{MAX}	Output frequency[2]	V_{VDD} = 3.0V, load = 10 pF	[5]			195	MHz
			[6]			348	MHz
		V_{VDD} = 3.0V, load = 30 pF	[5]			78	MHz
			[6]			149	MHz
t_{RISE}	Rise time[2]	V_{VDD} = 3.0V, load = 10 pF	[5]			2.21	ns
			[6]			1.26	ns
		V_{VDD} = 3.0V, load = 30 pF	[5]			5.45	ns
			[6]			2.88	ns
t_{FALL}	Fall time[2]	V_{VDD} = 3.0V, load = 10 pF	[5]			2.57	ns
			[6]			1.44	ns
		V_{VDD} = 3.0V, load = 30 pF	[5]			6.41	ns
			[6]			3.35	ns
I_{LEAK}	Input leakage current	Pull-up resistors disabled				1	µA
C_{IN}	Input capacitance,	[7]			2		pF
		PA09, PA10			16.5		pF
		PA11, PA12, PA18, PA19			18.5		pF
		PB14, PB15			5		pF

Notes:
1. V_{VDD} corresponds to either V_{VDDIN} or V_{VDDIO}, depending on the supply for the pin. Refer to Section 3.2 on page 8 for details.
2. These values are based on simulation and characterization of other AVR microcontrollers manufactured in the same process technology. These values are not covered by test limits in production.
3. This applies to all normal drive pads except PB13, PB17 and PB18.
4. This applies to PB13, PB17 and PB18 pads only.
5. This applies to all normal drive pad except PA00, PA01, PA02, PA03, PA04, PA05, PA06, PA07, PA08, PA09, PA10, PA11, PA12, PA13, PA18, PA19, PA27, PA30, PA31, PB13, PB16 and RESET_N.
6. This applies to PA00, PA01, PA02, PA03, PA04, PA05, PA06, PA07, PA08, PA09, PA10, PA11, PA12, PA13, PA18, PA19, PA27, PA30, PA31, PB13, PB16 and RESET_N pads only.
7. This applies to all normal drive pads except PA09, PA10, PA11, PA12, PA18, PA19, PB14, PB15.

Table 8-7. High-drive I/O Pin Characteristics[1]

Symbol	Parameter	Condition	Min	Typ	Max	Units
R_{PULLUP}	Pull-up resistance		9	15	25	kOhm
V_{IL}	Input low-level voltage	V_{VDD} = 3.0V	-0.3		+0.8	V
V_{IH}	Input high-level voltage	V_{VDD} = 3.6V	+2		V_{VDD} + 0.3	V
V_{OL}	Output low-level voltage	V_{VDD} = 3.0V, I_{OL} = 6mA			0.4	V

8.7 Oscillator Characteristics

8.7.1 Oscillator 0 (OSC0) Characteristics

8.7.1.1 Digital Clock Characteristics

The following table describes the characteristics for the oscillator when a digital clock is applied on XIN.

Table 8-10. Digital Clock Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
f_{CPXIN}	XIN clock frequency				50	MHz
t_{CPXIN}	XIN clock duty cycle		40		60	%

8.7.1.2 Crystal Oscillator Characteristics

The following table describes the characteristics for the oscillator when a crystal is connected between XIN and XOUT as shown in Figure 8-2. The user must choose a crystal oscillator where the crystal load capacitance C_L is within the range given in the table. The exact value of C_L can be found in the crystal datasheet. The capacitance of the external capacitors (C_{LEXT}) can then be computed as follows:

$$C_{LEXT} = 2(C_L - C_i) - C_{PCB}$$

where C_{PCB} is the capacitance of the PCB.

Table 8-11. Crystal Oscillator Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$1/(t_{CPMAIN})$	Crystal oscillator frequency		0.4		20	MHz
C_L	Crystal load capacitance		6		18	pF
C_i	Internal equivalent load capacitance			1.7		pF
$t_{STARTUP}$	Startup time	400 KHz Resonator SCIF.OSCCTRL.GAIN = 0[1]		198		µs
		2 MHz Quartz SCIF.OSCCTRL.GAIN = 0[1]		4666		
		8 MHz Quartz SCIF.OSCCTRL.GAIN = 1[1]		975		
		12 MHz Quartz SCIF.OSCCTRL.GAIN = 2[1]		615		
		16 MHz Quartz SCIF.OSCCTRL.GAIN = 2[1]		1106		
		20 MHz Quartz SCIF.OSCCTRL.GAIN = 3[1]		1109		

Notes: 1. Please refer to the SCIF chapter for details.

Table 8-18. Flash Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
T_{FPP}	Page programming time	f_{CLK_HSB}= 48MHz		5		ms
T_{FPE}	Page erase time			5		
T_{FFP}	Fuse programming time			1		
T_{FEA}	Full chip erase time (EA)			6		
T_{FCE}	JTAG chip erase time (CHIP_ERASE)	f_{CLK_HSB}= 115kHz		310		

Table 8-19. Flash Endurance and Data Retention

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
N_{FARRAY}	Array endurance (write/page)		100k			cycles
N_{FFUSE}	General Purpose fuses endurance (write/bit)		10k			cycles
t_{RET}	Data retention		15			years

8.9 Analog Characteristics

8.9.1 Voltage Regulator Characteristics

8.9.1.1 Electrical Characteristics

Table 8-20. Electrical Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{VDDIN}	Input voltage range		3	3.3	3.6	V
$V_{VDDCORE}$	Output voltage	V_{VDDIN} >= 3V	1.75	1.8	1.85	V
	Output voltage accuracy	I_{OUT} = 0.1mA to 100mA, V_{VDDIN}>3V		2		%
I_{OUT}	DC output current	V_{VDDIN}=3.3V			100	mA
I_{VREG}	Static current of internal regulator	Low power mode		10		µA

8.9.1.2 Decoupling Requirements

Table 8-21. Decoupling Requirements

Symbol	Parameter	Condition	Typ	Techno.	Units
C_{IN1}	Input regulator capacitor 1		1	NPO	nF

Table 8-21. Decoupling Requirements

Symbol	Parameter	Condition	Typ	Techno.	Units
C_{IN2}	Input regulator capacitor 2		4.7	X7R	nF
C_{OUT1}	Output regulator capacitor 1		470	NPO	nF
C_{OUT2}	Output regulator capacitor 2		2.2	X7R	µF

8.9.2 ADC Characteristics

Table 8-22. Channel Conversion Time and ADC Clock

Parameter	Conditions	Min.	Typ.	Max.	Unit
ADC Clock Frequency	10-bit resolution mode			5	MHz
	8-bit resolution mode			8	MHz
Startup Time	Return from Idle Mode			20	µs
Track and Hold Acquisition Time		600			ns
Conversion Time	ADC Clock = 5 MHz			2	µs
	ADC Clock = 8 MHz			1.25	µs
Throughput Rate	ADC Clock = 5 MHz			384 (1)	kSPS
	ADC Clock = 8 MHz			533 (2)	kSPS

1. Corresponds to 13 clock cycles: 3 clock cycles for track and hold acquisition time and 10 clock cycles for conversion.
2. Corresponds to 15 clock cycles: 5 clock cycles for track and hold acquisition time and 10 clock cycles for conversion.

Table 8-23. ADC Power Consumption

Parameter	Conditions	Min.	Typ.	Max.	Unit
Current Consumption on VDDANA (1)	On 13 samples with ADC clock = 5 MHz			1.25	mA

1. Including internal reference input current

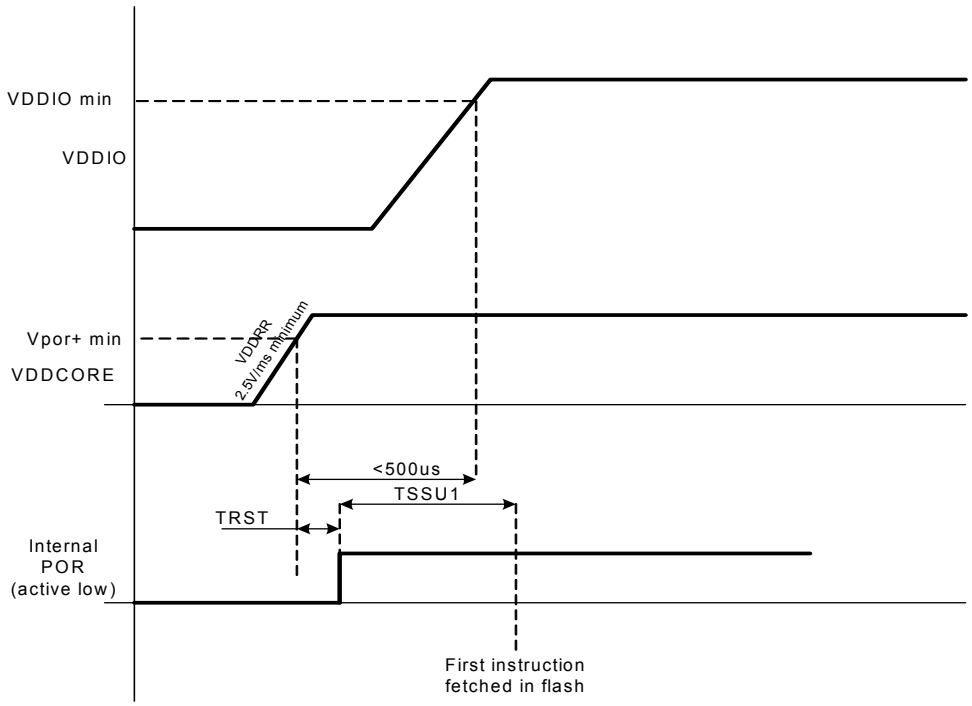
Table 8-24. Analog Inputs

Parameter	Conditions	Min.	Typ.	Max.	Unit
Input Voltage Range		0		VDDANA	V
Input Leakage Current				1	µA
Input Capacitance			7		pF
Input Resistance			370	810	Ohm

Table 8-25. Transfer Characteristics in 8-bit mode

Parameter	Conditions	Min.	Typ.	Max.	Unit
Resolution			8		Bit
Absolute Accuracy	ADC Clock = 5 MHz			0.8	LSB
	ADC Clock = 8 MHz			1.5	LSB
Integral Non-linearity	ADC Clock = 5 MHz		0.35	0.5	LSB
	ADC Clock = 8 MHz		0.5	1.0	LSB

Figure 8-6. Dual Supply Configuration



8.9.5 RESET_N Characteristics

Table 8-30. RESET_N Waveform Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
t_{RESET}	RESET_N minimum pulse width		10			ns

8.10 USB Transceiver Characteristics

8.10.1 Electrical Characteristics

Electrical Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
R_{EXT}	Recommended External USB Series Resistor	In series with each USB pin with ±5%		39		Ω

The USB on-chip buffers comply with the Universal Serial Bus (USB) v2.0 standard. All AC parameters related to these buffers can be found within the USB 2.0 electrical specifications.

Figure 9-3. QFN-48 Package Drawing

Table 9-8. Device and Package Maximum Weight

Weight	100 mg

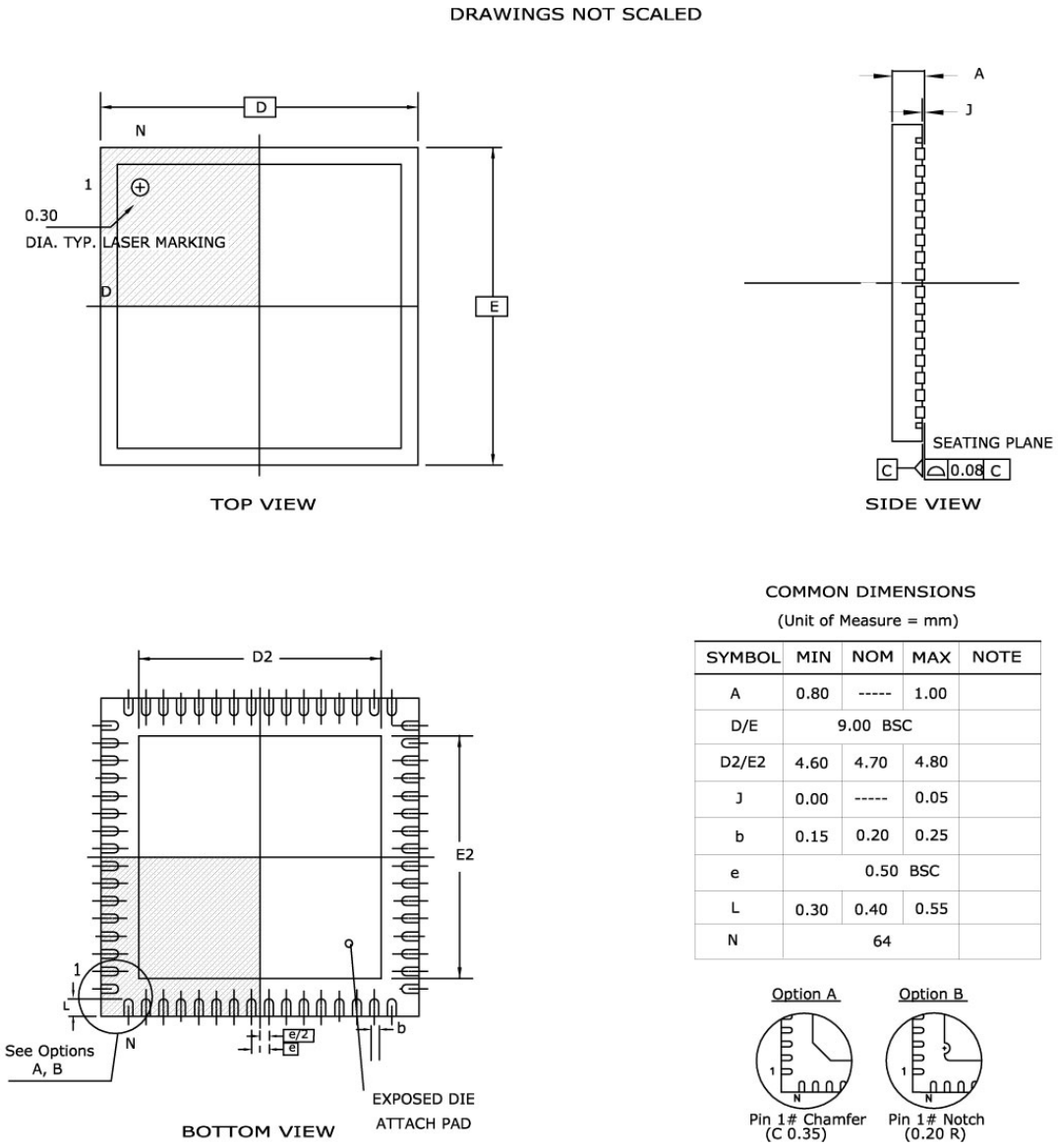
Table 9-9. Package Characteristics

Moisture Sensitivity Level	Jedec J-STD-20D-MSL3

Table 9-10. Package Reference

JEDEC Drawing Reference	M0-220
JESD97 Classification	e3

Figure 9-4. QFN-64 package drawing



SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	-----	1.00	
D/E		9.00 BSC		
D2/E2	4.60	4.70	4.80	
J	0.00	-----	0.05	
b	0.15	0.20	0.25	
e		0.50 BSC		
L	0.30	0.40	0.55	
N		64		

Notes : 1. This drawing is for general information only. Refer to JEDEC Drawing MO-220, Variation VMMD-4, for proper dimensions, tolerances, datums, etc.
2. Dimension b applies to metallized terminal and is measured between 0.15mm and 0.30mm from the terminal tip.
If the terminal has the optical radius on the other end of the terminal, the dimension should not be measured in that radius area.

Table 9-11. Device and Package Maximum Weight

Weight	200 mg

Table 9-12. Package Characteristics

Moisture Sensitivity Level	Jedec J-STD-20D-MSL3

Table 9-13. Package Reference

JEDEC Drawing Reference	M0-220
JESD97 Classification	e3

11.1.2 TWIS

1. **Clearing the NAK bit before the BTF bit is set locks up the TWI bus**
   When the TWIS is in transmit mode, clearing the NAK Received (NAK) bit of the Status Register (SR) before the end of the Acknowledge/Not Acknowledge cycle will cause the TWIS to attempt to continue transmitting data, thus locking up the bus.
   Fix/Workaround
   Clear SR.NAK only after the Byte Transfer Finished (BTF) bit of the same register has been set.

11.1.3 PWMA

1. **The SR.READY bit cannot be cleared by writing to SCR.READY**
   The Ready bit in the Status Register will not be cleared when writing a one to the corresponding bit in the Status Clear register. The Ready bit will be cleared when the Busy bit is set.
   Fix/Workaround
   Disable the Ready interrupt in the interrupt handler when receiving the interrupt. When an operation that triggers the Busy/Ready bit is started, wait until the ready bit is low in the Status Register before enabling the interrupt.

11.2 Rev. B

11.2.1 Power Manager

1. **TWIS may not wake the device from sleep mode**
   If the CPU is put to a sleep mode (except Idle and Frozen) directly after a TWI Start condition, the CPU may not wake upon a TWIS address match. The request is NACKed.
   Fix/Workaround
   When using the TWI address match to wake the device from sleep, do not switch to sleep modes deeper than Frozen. Another solution is to enable asynchronous EIC wake on the TWIS clock (TWCK) or TWIS data (TWD) pins, in order to wake the system up on bus events.

11.2.2 SPI

1. **SPI disable does not work in SLAVE mode**
   SPI disable does not work in SLAVE mode.
   Fix/Workaround
   Read the last received data, then perform a software reset by writing a one to the Software Reset bit in the Control Register (CR.SWRST).

2. **PCS field in receive data register is inaccurate**
   The PCS field in the SPI_RDR register does not accurately indicate from which slave the received data is read.
   Fix/Workaround
   None.

3. **SPI data transfer hangs with CSR0.CSAAT==1 and MR.MODFDIS==0**
   When CSR0.CSAAT==1 and mode fault detection is enabled (MR.MODFDIS==0), the SPI module will not start a data transfer.
   Fix/Workaround
   Disable mode fault detection by writing a one to MR.MODFDIS.

4. **Disabling SPI has no effect on the SR.TDRE bit**
   Disabling SPI has no effect on the SR.TDRE bit whereas the write data command is filtered when SPI is disabled. Writing to TDR when SPI is disabled will not clear SR.TDRE. If SPI is disabled during a PDCA transfer, the PDCA will continue to write data to TDR until its buffer is empty, and this data will be lost.
   Fix/Workaround
   Disable the PDCA, add two NOPs, and disable the SPI. To continue the transfer, enable the SPI and PDCA.

5. **SPI bad serial clock generation on 2nd chip_select when SCBR=1, CPOL=1, and NCPHA=0**
   When multiple chip selects (CS) are in use, if one of the baudrates equal 1 while one (CSRn.SCBR=1) of the others do not equal 1, and CSRn.CPOL=1 and CSRn.NCPHA=0, then an additional pulse will be generated on SCK.
   Fix/Workaround
   When multiple CS are in use, if one of the baudrates equals 1, the others must also equal 1 if CSRn.CPOL=1 and CSRn.NCPHA=0.

6. **Timer Counter**

7. **Channel chaining skips first pulse for upper channel**
   When chaining two channels using the Block Mode Register, the first pulse of the clock between the channels is skipped.
   Fix/Workaround
   Configure the lower channel with RA = 0x1 and RC = 0x2 to produce a dummy clock cycle for the upper channel. After the dummy cycle has been generated, indicated by the SR.CPCS bit, reconfigure the RA and RC registers for the lower channel with the real values.

11.2.3 TWIS

1. **Clearing the NAK bit before the BTF bit is set locks up the TWI bus**
   When the TWIS is in transmit mode, clearing the NAK Received (NAK) bit of the Status Register (SR) before the end of the Acknowledge/Not Acknowledge cycle will cause the TWIS to attempt to continue transmitting data, thus locking up the bus.
   Fix/Workaround
   Clear SR.NAK only after the Byte Transfer Finished (BTF) bit of the same register has been set.

11.2.4 PWMA

1. **The SR.READY bit cannot be cleared by writing to SCR.READY**
   The Ready bit in the Status Register will not be cleared when writing a one to the corresponding bit in the Status Clear register. The Ready bit will be cleared when the Busy bit is set.
   Fix/Workaround
   Disable the Ready interrupt in the interrupt handler when receiving the interrupt. When an operation that triggers the Busy/Ready bit is started, wait until the ready bit is low in the Status Register before enabling the interrupt.

11.3 Rev. A

11.3.1 GPIO

1. **Clearing Interrupt flags can mask other interrupts**
   When clearing interrupt flags in a GPIO port, interrupts on other pins of that port, happening in the same clock cycle will not be registered.
   Fix/Workaround
   Read the PVR register of the port before and after clearing the interrupt to see if any pin change has happened while clearing the interrupt. If any change occurred in the PVR between the reads, they must be treated as an interrupt.

11.3.2 Power Manager

1. **Clock Failure Detector (CFD) can be issued while turning off the CFD**
   While turning off the CFD, the CFD bit in the Status Register (SR) can be set. This will change the main clock source to RCSYS.
   Fix/Workaround
   Solution 1: Enable CFD interrupt. If CFD interrupt is issues after turning off the CFD, switch back to original main clock source.
   Solution 2: Only turn off the CFD while running the main clock on RCSYS.

2. **Requesting clocks in idle sleep modes will mask all other PB clocks than the requested**
   In idle or frozen sleep mode, all the PB clocks will be frozen if the TWIS or the AST needs to wake the CPU up.
   Fix/Workaround
   Disable the TWIS or the AST before entering idle or frozen sleep mode.

3. **SPI**

4. **SPI disable does not work in SLAVE mode**
   SPI disable does not work in SLAVE mode.
   Fix/Workaround
   Read the last received data, then perform a software reset by writing a one to the Software Reset bit in the Control Register (CR.SWRST).

5. **PCS field in receive data register is inaccurate**
   The PCS field in the SPI_RDR register does not accurately indicate from which slave the received data is read.
   Fix/Workaround
   None.

6. **SPI data transfer hangs with CSR0.CSAAT==1 and MR.MODFDIS==0**
   When CSR0.CSAAT==1 and mode fault detection is enabled (MR.MODFDIS==0), the SPI module will not start a data transfer.
   Fix/Workaround
   Disable mode fault detection by writing a one to MR.MODFDIS.

7. **Disabling SPI has no effect on the SR.TDRE bit**
   Disabling SPI has no effect on the SR.TDRE bit whereas the write data command is filtered when SPI is disabled. Writing to TDR when SPI is disabled will not clear SR.TDRE. If SPI is disabled during a PDCA transfer, the PDCA will continue to write data to TDR until its buffer is empty, and this data will be lost.

11.3.3 Timer Counter

1. **Channel chaining skips first pulse for upper channel**
   When chaining two channels using the Block Mode Register, the first pulse of the clock between the channels is skipped.
   Fix/Workaround
   Configure the lower channel with RA = 0x1 and RC = 0x2 to produce a dummy clock cycle for the upper channel. After the dummy cycle has been generated, indicated by the SR.CPCS bit, reconfigure the RA and RC registers for the lower channel with the real values.

11.3.4 TWIS

1. **TWIS stretch on Address match error**
   When the TWIS stretches TWCK due to a slave address match, it also holds TWD low for the same duration if it is to be receiving data. When TWIS releases TWCK, it releases TWD at the same time. This can cause a TWI timing violation.
   Fix/Workaround
   None.

2. **Clearing the NAK bit before the BTF bit is set locks up the TWI bus**
   When the TWIS is in transmit mode, clearing the NAK Received (NAK) bit of the Status Register (SR) before the end of the Acknowledge/Not Acknowledge cycle will cause the TWIS to attempt to continue transmitting data, thus locking up the bus.
   Fix/Workaround
   Clear SR.NAK only after the Byte Transfer Finished (BTF) bit of the same register has been set.

3. **CAT**

4. **CAT module does not terminate QTouch burst on detect**
   The CAT module does not terminate a QTouch burst when the detection voltage is reached on the sense capacitor. This can cause the sense capacitor to be charged more than necessary. Depending on the dielectric absorption characteristics of the capacitor, this can lead to unstable measurements.
   Fix/Workaround
   Use the minimum possible value for the MAX field in the ATCFG1, TG0CFG1, and TG1CFG1 registers.

11.3.5 PWMA

1. **The SR.READY bit cannot be cleared by writing to SCR.READY**
   The Ready bit in the Status Register will not be cleared when writing a one to the corresponding bit in the Status Clear register. The Ready bit will be cleared when the Busy bit is set.
   Fix/Workaround
   Disable the Ready interrupt in the interrupt handler when receiving the interrupt. When an operation that triggers the Busy/Ready bit is started, wait until the ready bit is low in the Status Register before enabling the interrupt.

2.

12. Datasheet Revision History

Please note that the referring page numbers in this section are referred to this document. The referring revision in this section are referring to the document revision.

12.1 Rev. A – 11/2009

1. Initial revision.

12.2 Rev. B – 04/2011

1. Minor.

12.3 Rev. C – 07/2011

1. Final revision.

12.4 Rev. D – 11/2011

1. Adding errata for silicon Revision C .
2. Fixed PLLOPT field description in SCIF chapter